

CLAIMS

What is claimed is:

- 1 1. A method for handling network accounting information, comprising:
 - 2 (a) receiving records indicative of network events from an input source;
 - 3 (b) selecting action events based on the input source; and
 - 4 (c) executing the selected action events on the records.
- 1 2. The method as recited in claim 1, wherein the action events include computer
2 code for executing a process using the records.
- 1 3. The method as recited in claim 2, and further comprising the step of
2 compiling the computer code prior to the execution thereof.
- 1 4. The method as recited in claim 1, and further comprising the step of storing
2 data associated with the records.
- 1 5. The method as recited in claim 4, wherein the data is stored in a table.
- 1 6. The method as recited in claim 5, wherein the table includes a plurality of
2 rows each containing a plurality of columns each including data of a different
3 type.
- 1 7. The method as recited in claim 6, wherein the data of each of the rows
2 expires after a predetermined time period.

- 1 8. The method as recited in claim 7, wherein an action event is executed to
2 determine whether the data of each of the rows is deleted upon expiring.
- 1 9. The method as recited in claim 1, wherein multiple action events are
2 executed in parallel.
- 1 10. A computer program product for handling network accounting information,
2 comprising:
3 (a) computer code for receiving records indicative of network events from an
4 input source;
5 (b) computer code for selecting action events based on the input source; and
6 (c) computer code for executing the selected action events on the records.
- 1 11. The computer program product as recited in claim 10, wherein the action
2 events include computer code for executing a process using the records.
- 1 12. The computer program product as recited in claim 11, and further comprising
2 computer code for compiling the computer code prior to the execution
3 thereof.
- 1 13. The computer program product as recited in claim 10, and further comprising
2 computer code for storing data associated with the records.
- 1 14. The computer program product as recited in claim 13, wherein the data is
2 stored in a table.
- 1 15. The computer program product as recited in claim 14, wherein the table
2 includes a plurality of rows each containing a plurality of columns each
3 including data of a different type.

- 1 16. The computer program product as recited in claim 15, wherein the data of
2 each of the rows expires after a predetermined time period.
- 1 17. The computer program product as recited in claim 16, wherein an action
2 event is executed to determine whether the data of each of the rows is deleted
3 upon expiring.
- 1 18. The computer program product as recited in claim 10, wherein multiple
2 action events are executed in parallel.
- 1 19. A system for handling network accounting information, comprising:
2 (a) logic for receiving records indicative of network events from an input source;
3 (b) logic for selecting action events based on the input source; and
4 (c) logic for executing the selected action events on the records.
- 1 20. A method for handling network accounting information, comprising:
2 (a) receiving records indicative of network events from an input source;
3 (b) storing data associated with the records in a table, wherein the table includes
4 a plurality of rows each containing a plurality of columns each including data
5 of a different type, the data of each of the rows expiring after a predetermined
6 time period;
7 (c) selecting action events based on the input source; and
8 (d) executing the selected action events on the records;
9 (e) wherein at least one of the action events is executed to delete the data of each
10 of the rows upon expiring.
- 1 21. A method for handling network accounting information of any type,
2 comprising:
3 (a) reading configuration data which defines a table by specifying at least one
4 field identifier and a timeout type and period, the configuration data further
5 defining a plurality of input sources by specifying at least one parameter for

- 6 each input source, the configuration data further defining a plurality of action
7 events by specifying code capable of executing each action event;
8 (b) creating the table defined by the field identifier of the configuration data;
9 (c) initializing the input sources;
10 (d) loading event handlers with the code included with the configuration data;
11 (e) receiving records indicative of network events from the initialized input
12 sources;
13 (f) storing the records in the table;
14 (g) selecting action events based on the input source associated with the received
15 records;
16 (h) executing the selected action events on the records utilizing the event
17 handlers; and
18 (i) deleting the records upon expiring in accordance with the timeout type and
19 period of the configuration data;
20 (j) wherein at least one of the action events is executed to determine whether the
21 data of each of the rows is deleted upon expiring.

1 22. The method as recited in claim 21, wherein the execution of the selected
2 action events includes: discarding records stored during the execution of
3 previous action events, parsing the configuration data associated with the
4 selected action events, and utilizing the parsed configuration data to repeat
5 steps (b) through (d).

- 1 23. A data structure for handling network accounting information of any type,
2 comprising:
3 (a) a configuration data object which defines a table by specifying at least one
4 field identifier and a timeout type and period, the configuration data object
5 further defining a plurality of input sources by specifying at least one
6 parameter for each input source, the configuration data object further
7 defining a plurality of action events by specifying code capable of executing
8 each action event;

9 (b) wherein the configuration data object is adapted for being used to create the
10 table defined by the field identifier of the configuration data object, initialize
11 the input sources, and load event handlers with the code included with the
12 configuration data object.

[illegible]